CLAIMS

I/We claim:

[c1] 1. A method in a computer system for returning a stream to a task executing an operating system call that is blocked, the computer system having a processor with multiple streams, the method comprising:

under control of the operating system executing on a stream, invoking a function provided by the task;

under control of the invoked function, executing instructions of the task on that stream; and

under control of the operating system, notifying the task when the operating system call is complete.

[c2] 2. The method of claim 1 wherein the notifying includes

invoking a function provided by the task using a stream of the operating system; and

under control of that invoked function,

indicating that the operating system call is complete; and

invoking another operating system call to return the operating system stream to the operating system.

[c3] 3. The method of claim 1 wherein the executing of instructions on that stream includes

indicating that a thread that invoked the operating system call is blocked; and

executing another thread on that stream.

[c4] 4. A system for returning a stream to a task executing an operating system call that is blocked, the system having a processor with multiple streams and comprising:

a component that, under control of the operating system executing on a stream, invokes a function provided by the task;

a component that, under control of the invoked function, executes instructions of the task on that stream; and

a component that, under control of the operating system, notifies the task when the operating system call is complete.

[c5] 5. The system of claim 4 wherein the notification includes

invoking a function provided by the task using a stream of the operating system; and

under control of that invoked function,

indicating that the operating system call is complete; and

invoking another operating system call to return the operating system stream to the operating system.

[c6] 6. The system of claim 4 wherein the instructions of the test on that stream include

an indication that a thread that invoked the operating system call is blocked; and

execution of another thread on that stream.

[c7] 7. A method in a computer system for assigning a processor resource to a thread of a task, the method comprising:

under control of a thread of the task, invoking an operating system call that will block and wait for the occurrence of an event; and

under control of the operating system, when the call is blocked, invoking a routine of the task so that the routine can assign the processor resource to another thread of the task.

[c8] 8. The method of claim 7 wherein the processor resource is a stream of a processor that supports multiple streams.

[c9] 9. The method of claim 7 wherein the task registers the routine with the operating prior to invoking the operating system call.

[c10] 10. The method of claim 7 including notifying the task when a operating system call completes.

[c11] 11. A system for assigning a processor resource to a thread of a task, the system comprising:

a component for under control of a thread of the task, invoking an operating system call that will block and wait for the occurrence of an event; and

a component for, under control of the operating system, invoking a routine of the task so that the routine can assign the processor resource to another thread of the task.

[c12] 12. The system of claim 11 wherein the processor resource is a stream of a processor that supports multiple streams.

[c13] 13. The system of claim 11 wherein the task registers the routine with the operating prior to invoking the operating system call.

[c14] 14. The system of claim 11 including notifying the task when a operating system call completes.

[c15] 15. A method in a computer system for returning a stream to a user program, the computer system having an operating system, the method comprising:

under control of the operating system,

when an operating system call in a stream will block, invoking a first function of a task that will return the stream to the task; and

when the operating system call becomes unblocked, invoking a second function of the task.

[c16] 16. The method of claim 15 wherein the operating system invokes the first function using the stream that will block.

[c17] 17. The method of claim 16 wherein invoking the first function returns the stream to the user program.

[c18] 18. The method of claim 17 wherein the user program selects a thread that is not blocked for execution on the stream.

[c19] 19. The method of claim 15 wherein the second function schedules for restarting a thread that was blocked on the operating system call that was blocked.

[c20] 20. The method of claim 15 wherein the second function returns a stream provided by the operating system.

[c21] 21. A method in a computer system for returning a stream to a user program, the computer system having an operating system, the method comprising:

under control of the user program, invoking an operating system call;

executing the operating system call in a user stream of the user program; and

under control of the operating system, when the operating system call will block,

when a thread making the operating system call is locked, waiting for the operating system call to become unblocked; and

when a thread making the operating system call is not locked,

invoking a first function of the user program that will return the stream to the task;

under control of a trap handler routine, placing the thread in a blocked pool and selecting another thread to execute on the stream; and

when the operating system call becomes unblocked, invoking a second function of the user program in a stream of the operating system.

[c22] 22. The method of claim 21 wherein the second function schedules for restarting a thread that was blocked on the operating system call that was blocked.

[c23] 23. The method of claim 21 wherein the second function returns a stream provided by the operating system.